

# COMMODORE MPS-802 DOT MATRIX PRINTER

A Friendly Introduction to Your MPS-802 Dot Matrix Printer



USER'S GUIDE

 **commodore**  
COMPUTER

## INFORMATION TO USER

"WARNING: THIS EQUIPMENT HAS BEEN CERTIFIED TO COMPLY WITH THE LIMITS FOR A CLASS B COMPUTING DEVICE, PURSUANT TO SUBPART J OF PART 15 OF FCC RULES. ONLY PERIPHERALS (COMPUTER INPUT/OUTPUT DEVICES, TERMINALS, PRINTERS, ETC.) CERTIFIED TO COMPLY WITH THE CLASS B LIMITS MAY BE ATTACHED TO THIS COMPUTER. OPERATION WITH NON-CERTIFIED PERIPHERALS IS LIKELY TO RESULT IN INTERFERENCE TO RADIO AND TV RECEPTION."

"THIS EQUIPMENT GENERATES AND USES RADIO FREQUENCY ENERGY AND IF NOT INSTALLED PROPERLY, THAT IS, IN STRICT ACCORDANCE WITH THE MANUFACTURER'S INSTRUCTIONS, MAY CAUSE INTERFERENCE TO RADIO AND TELEVISION RECEPTION. IT HAS BEEN TYPE TESTED AND FOUND TO COMPLY WITH THE LIMITS FOR A CLASS B COMPUTING DEVICE IN ACCORDANCE WITH THE SPECIFICATIONS IN SUBPART J OF PART 15 OF FCC RULES, WHICH ARE DESIGNED TO PROVIDE REASONABLE PROTECTION AGAINST SUCH INTERFERENCE IN A RESIDENTIAL INSTALLATION. HOWEVER, THERE IS NO GUARANTEE THAT INTERFERENCE WILL NOT OCCUR IN A PARTICULAR INSTALLATION. IF THIS EQUIPMENT DOES CAUSE INTERFERENCE TO RADIO OR TELEVISION RECEPTION, WHICH CAN BE DETERMINED BY TURNING THE EQUIPMENT OFF AND ON, THE USER IS ENCOURAGED TO TRY TO CORRECT THE INTERFERENCE BY ONE OR MORE OF THE FOLLOWING MEASURES:

- REORIENT THE RECEIVING ANTENNA
- RELOCATE THE COMPUTER WITH RESPECT TO THE RECEIVER
- MOVE THE COMPUTER AWAY FROM THE RECEIVER
- PLUG THE COMPUTER INTO A DIFFERENT OUTLET SO THAT COMPUTER AND RECEIVER ARE ON DIFFERENT BRANCH CIRCUITS

"IF NECESSARY, THE USER SHOULD CONSULT THE DEALER OR AN EXPERIENCED RADIO/TELEVISION TECHNICIAN FOR ADDITIONAL SUGGESTIONS. THE USER MAY FIND THE FOLLOWING BOOKLET PREPARED BY THE FEDERAL COMMUNICATIONS COMMISSION HELPFUL: 'HOW TO IDENTIFY AND RESOLVE RADIO-TV INTERFERENCE PROBLEMS.' THIS BOOKLET IS AVAILABLE FROM THE U.S. GOVERNMENT PRINTING OFFICE, WASHINGTON, D.C. 20540, STOCK NO. 004-000-00345-4."

PART NO: 320970

# **COMMODORE MPS-802 DOT MATRIX PRINTER USER'S GUIDE**

**A Friendly Introduction to Your MPS-802 Dot Matrix Printer**

 **commodore**  
COMPUTER



# Table Of Contents

Page No.

<b>Section 1: GENERAL INFORMATION</b> .....	1
DESCRIPTION .....	1
Printing Characteristics .....	1
Interface .....	1
Paper Feed Mechanisms .....	2
Internal Microprocessor System .....	2
Specifications .....	2
UNPACKING YOUR PRINTER .....	3
<b>Section 2: PREPARING TO USE YOUR PRINTER</b> .....	4
CONNECTING THE PRINTER TO YOUR COMPUTER .....	6
INSTALLING THE PAPER .....	6
PERFORMING THE POWER-ON TEST .....	6
PERFORMING THE PRINT HEAD TEST .....	7
<b>Section 3: USING YOUR PRINTER</b> .....	8
SPECIAL PRINTER-ASSOCIATED COMMANDS .....	9
The OPEN Command .....	9
The CMD Command .....	9
The PRINT # Command .....	10
The CLOSE Command .....	10
PRINTING IN THE DIRECT MODE .....	11
PRINTING UNDER PROGRAM CONTROL .....	11
FORMAT CONTROL .....	15
Printing Data Exactly as Received: <b>sa = 0</b> .....	16
Printing Data According to a Previously Defined Format: <b>sa = 1</b> .....	16
Storing the Formatting Data: <b>sa = 2</b> .....	17
The Formatting Characters .....	18
Numeric: .....	18
Alpha (String Data): .....	19
Skip (Blank): .....	20
Literals in Format Strings .....	20
Setting the Number of Lines Per Page: <b>sa = 3</b> .....	21
Enabling the Printer Diagnostic Messages: <b>sa = 4</b> .....	21
Defining a Programmable Character: <b>sa = 5</b> .....	23
Setting Spacing Between Lines: <b>sa = 6</b> .....	25
Selecting Upper/Lowercase: <b>sa = 7</b> .....	25
Suppressing Diagnostic Message Printing: <b>sa = 9</b> .....	26
Resetting the Printer: <b>sa = 10</b> .....	26
SPECIAL CHARACTER FUNCTIONS .....	26
Enhanced Characters .....	26
Paging .....	27
Specifying Individual Upper- and Lowercase Characters .....	27
Reversing a Field .....	27
The Carriage Return .....	28
Quotation Marks .....	28
APPENDIX .....	
SIMPLE STRING FORMATTING .....	29
LEADING ZEROES FORCED .....	31
SIGNED NUMBERS .....	32
DECIMAL FRACTION SPECIFICATION .....	33
FIXED AND FLOATING \$ FIELDS .....	34
MIXED FORMAT FIELD TYPES .....	35
FORMATTING WITH LITERALS .....	36

## **Section 1**

# **GENERAL INFORMATION**

The FRICTION/SPROCKET FEED MPS-802 Printer adds a great deal of versatility and convenience to the use of your Commodore Computer. While this manual contains all the information you need to check out, connect, and operate your printer, you should also refer to other Commodore manuals to get the most out of your computer system.

### **DESCRIPTION**

The MPS-802 printer is designed to operate through software control. It prints upper- and lower-case alphabetic characters, numeric characters, and all the graphic characters available on your Commodore computer, and even a custom user-defined character. In addition, your printer has considerable formatting capability owing to its internal microprocessor system.

### **Printing Characteristics**

Your printer employs a dot matrix print head. The wire heavy-duty jeweled head has a life expectancy of 50 million characters. If a print head malfunction should occur, obtain authorized technical assistance. Failure to do this may void warranty.

### **Interface**

Your printer is designed to connect directly into your computer through the Serial Port (6-pins). However, you can connect your printer to as many as 4 VIC disk drives by daisy-chaining. Daisy-chaining means connecting 1 peripheral to your computer and plugging additional peripherals into the Serial Port of the last item connected. For more information about Serial Port specifications, please refer to the Serial Bus Section in your Programmer's Reference Guide.

## Paper Feed Mechanism

The MPS-802 Printer has a FRICTION/SPROCKET feed mechanism that uses sprocket holes to hold the paper edges. This model is especially useful for printing business forms. See Figure 1.

Normal paper without sprocket holes can be used, too.

## Internal Microprocessor System

Your printer contains a microprocessor system that resets and executes a diagnostic and initialization sequence when you turn on the power. In addition, it contains a random access memory (RAM) in which you can store formatting data. Because your printer is an "intelligent" peripheral, it uses none of your computer's memory.

## Specifications

Table 1 shows the specifications for the MPS-802 Printer.

TABLE 1

Model 1526 Printer Specifications	
Printing Method	Serial Impact Dot Matrix
Print Rate	45 lpm* with 80 columns printed 78 lpm with 40 columns printed 124 lpm with 20 columns printed
Print Direction	Bi-directional
Column Capacity	80
Character Font	8 X 8
Line Spacing	Programmable
Character Size	0.094" high, 0.08" wide
Copies	3, including original
Ribbon Type	Cartridge
Ribbon Life	1.2 X 10 <sup>6</sup> characters
Ribbon Cartridge	Commodore P/N 613160550
Paper Width	4.5" to 10" (including tractor holes)
Forms	7.5 + (0.5 X 2 sprocket margins) Pin-to-pin distance: .5" longitudinally 9.5" laterally 5/32" diameter

\*Lines per minute

## **UNPACKING YOUR PRINTER**

Before you unpack your printer, inspect the shipping carton for signs of damage. If it appears to be damaged, be especially careful when you inspect its contents. DON'T throw away any of the packaging material until you have located all the contents of the carton! The package should contain:

1. Commodore Printer, Model MPS-802
2. Serial Cable
3. User's Manual
4. Warranty card
5. Printer ribbon cartridge

If any of these is missing or damaged, notify your Commodore dealer immediately.

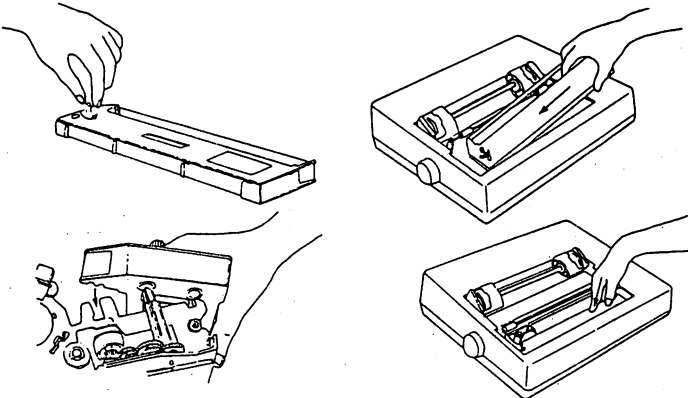


## Section 2

# PREPARING TO USE YOUR PRINTER

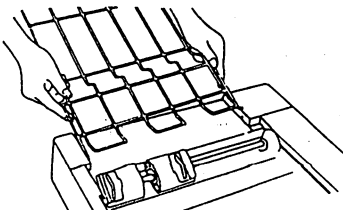
Before starting to use your printer, you should make sure that it is in working condition. This procedure includes checking for obstructions in the path of the print head or paper feed and making sure that the printer ribbon is properly in place. Follow this procedure:

1. Lift the plastic cover gently to expose print head and mechanism.
2. Carefully remove any foreign material that may have fallen into the mechanism.
3. Remove the ribbon cartridge from its box and turn the plastic knob counterclockwise so the ribbon is tight.
4. Hold the ribbon cartridge with the plastic knob at the top left side, then set it on the two side frames of the printer mechanism with tilting the cartridge so the two front hooks on the side frames be engaged with two catches on both left and right sides of bottom of cartridge, then steer down while pressing the ribbon side so the two side tabs of cartridge are positioned into the slots on the side frames of the printer mechanism.
5. Turn the plastic knob on cartridge in the direction of the arrow to take up the slack in the ribbon. Continue to turn the knob until the ribbon slips into front of printer head by itself.

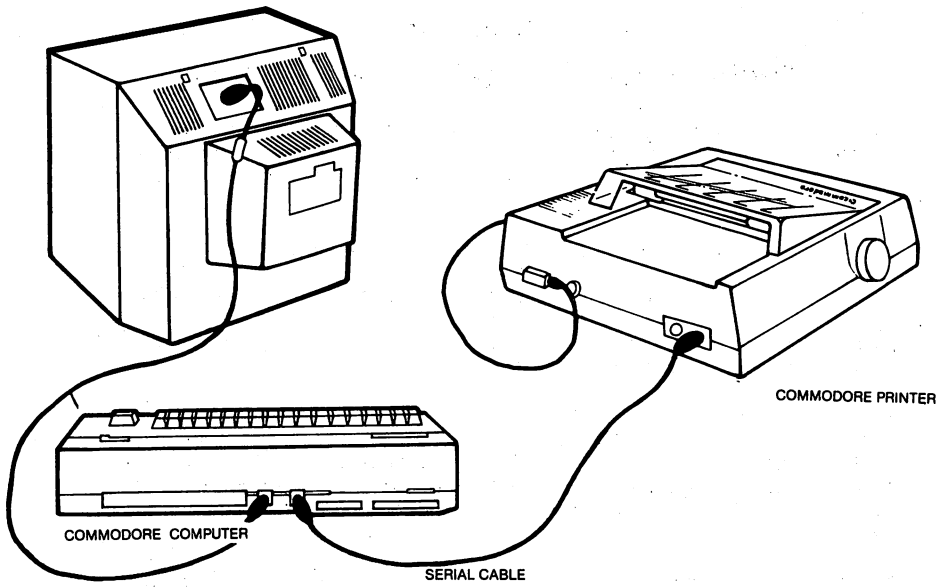


**Figure 2A. The Ribbon Cartridge**

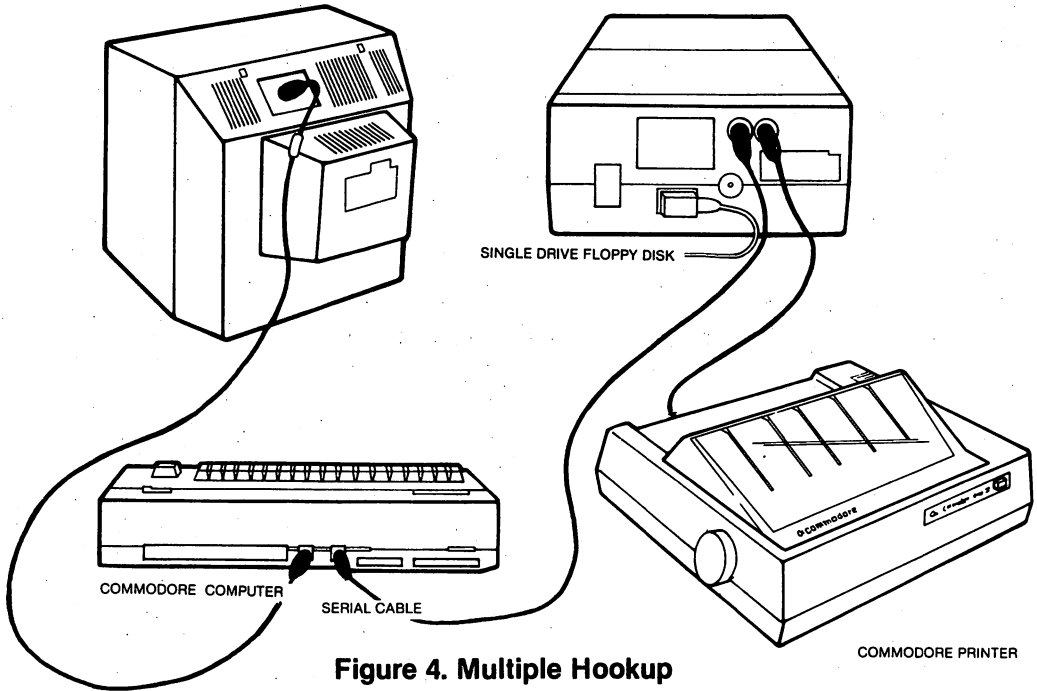
6. Replace the plastic cover.
7. Insert the wire paper holder in the two holes in the rear of the cabinet as shown in figure 2B.



**Figure 2B. Wire Paper Holder**



**Figure 3. Printer to Computer Hookup**



**Figure 4. Multiple Hookup**

## CONNECTING THE PRINTER TO YOUR COMPUTER

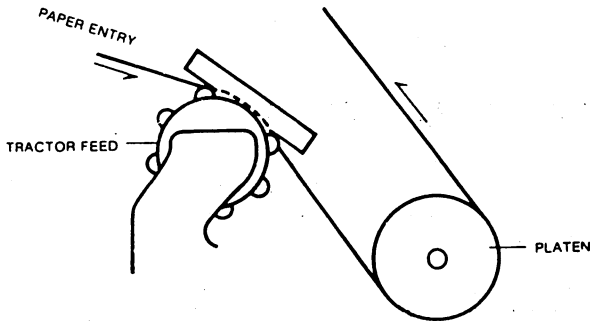
To connect your printer to your Commodore computer, please follow the instructions in the order listed below.

- a. Make sure that both your computer and printer are turned OFF.
- b. Connect one end of the 6-pin DIN cable to either of the two connector sockets located in the back of your printer at the lower left. This cable is "keyed" so that you cannot plug the cable in the wrong way. This means that the pins should be positioned so that a slight pressure will insert the cable properly. You should not have to force these pins in, and doing so could damage your cable.
- c. Connect the other end of the cable to your computer in the Serial Port Connector located in the back of your computer. Make sure that you are "keying" the pins properly into the connector with 6 holes.
- d. Now plug the printer's power cord into a standard AC wall outlet. However, do NOT turn the equipment on yet.

## INSTALLING THE PAPER

The tractor feed printer accepts standard fan-folded pin feed paper. You can purchase this paper from your local Commodore dealer.

The paper drive in this printer may be adjusted to accept varying widths of paper. This is done by sliding the pin feed mechanism to the desired position. Depress the paper feed button in the upper right-hand top panel to advance the paper after it is loaded. To remove the paper, open the feed mechanism retainers and pull the paper out.



**Figure 5. Paper Path**

**CAUTION : DO NOT FEED PAPER BACKWARDS.**



## Section 3

# USING YOUR PRINTER

Your printer does much more than just give you clean, fast copy. Since it has its own internal microprocessor system, it is very versatile. In this section you will learn how to use your printer to print out listings, program results, and graphic displays. You will learn how to use it to format data and enhance character size. You will even be able to create your own custom characters.

Before you attempt to use your printer, make sure you know how to do the following:

1. Operate your Commodore computer.
2. Do elementary programming in BASIC.
3. Write files to and from a peripheral device such as the DATASSETTE\* recorder or the Commodore Floppy Disk Drive.
4. Open and close files.

You should refer to your computer User Manual for this information.

This section uses these conventions to indicate certain actions or requirements.

### Example

### Description

*dn*

Italicized lowercase letters in a syntax indicate that you should enter something (a variable) in place of the letters.

[ ]

Brackets indicate optional usage.

NOTE: The commands described in this manual apply specifically to use with the printer. Certain commands may follow a slightly different general form or produce different results from those described here when they are used for the computer or other peripheral devices. Consult the appropriate manual for the exact usage of these commands in other contexts.

\*DATASSETTE is a trademark of Commodore Electronics, Ltd.

## PRINTER-ASSOCIATED COMMANDS

### The OPEN Command

The syntax of the OPEN command is:

**OPEN *lfn, dn, [sa]***

This command sets a correspondence between a file number and a physical device. The *lfn* or *logical file* may be any number you choose to assign to your file from 1 to 255. It doesn't matter which number you choose as long as you remain consistent throughout your set of commands. The *dn* or *device number* (also known as the primary address) refers to the device to which you wish to send the file. In the case of the printer, the number must be 4 since that is the number assigned to it at the factory.

NOTE: If you want to change a printer device number, you can have it changed by an authorized Commodore technician. See your Commodore dealer about this. (You might want to have a printer's device number changed if you have two printers attached to your computer. You need to refer to each of them individually; hence, the need to change one of the device numbers.)

The *sa* or *secondary address* is used to specify a particular printer function. Secondary addresses are thoroughly discussed in the section entitled FORMAT CONTROL.

### The CMD Command

The syntax of this command is:

**CMD *lfn***

CMD transfers output from the screen to the printer. The *lfn* must be the same as in the OPEN statement with which it is associated. Unlike a PRINT command, the line or bus to the receiving device is left open. The line or bus to the receiving device (in this case, the printer) is said to be "listening." If you follow a CMD command with a PRINT or a LIST command, the output is directed to the printer instead of the screen.

## The PRINT # Command

The syntax of the PRINT # command is:

**PRINT # *lfn*, data**

PRINT # works just like PRINT except that it directs output to the printer instead of the video screen. After printing the designated data, the line or bus to the printer is said to be "unlistened." Therefore, if you have used the CMD command, it is necessary to follow it with a PRINT # command in order to "unlisten" the connection between the printer and the computer.

NOTE: In standard Commodore BASIC, the PRINT command can be abbreviated as a question mark (?). You may not do this with PRINT #. It must always be typed out as PRINT #.

## The CLOSE Command

The syntax of this command is:

**CLOSE *lfn***

You should always close a file after printing to it. You may not exceed ten open files, so you should make a habit of closing files when you are finished with them. This way you will always have the maximum number of files available for use.

Remember when using the CMD command, since CMD does not "unlisten" the line to the printer, you must always precede a CLOSE command with PRINT # in order to properly close the file.

### Example:

Right		Wrong
OPEN 5,4 PRINT #5,"HELLO THERE" CLOSE 5		
OPEN 5,4 CMD 5,"HELLO THERE" PRINT #5;CLOSE 5	not	OPEN 5,4 CMD 5,"HELLO THERE" CLOSE 5
OPEN 5,4 CMD 5,"HELLO THERE" PRINT #5,"HELLO THERE" CLOSE 5	not	OPEN 5,4 CMD 5,"HELLO THERE" PRINT #5,"HELLO THERE" PRINT #5: CLOSE 5
OPEN 5,4 PRINT #5,"HELLO THERE" CMD 5,"HELLO THERE" PRINT #5:CLOSE 5	not	OPEN 5,4 PRINT #5,"HELLO THERE" CMD 5,"HELLO THERE" CLOSE 5

Armed with these thumbnail descriptions of what the printer-related commands do, you can now proceed to the next part of this section, which tells you how to use these commands to control your printer. You can find more detailed information about these commands in your computer User Manual.

## PRINTING IN THE DIRECT MODE

The direct mode of communication allows you to enter printing commands at the keyboard.

The following example shows the entire direct mode listing process of a short program.

### Example:

You type:	The screen displays:	The printer prints:	Comments:
OPEN 3,4	OPEN 3,4 READY.		You open the file and give it a <i>lfn</i> of 3. The 4 makes the file available to the printer.
CMD 3	CMD 3	READY.	The printer is "listening."
LIST	LIST		The program is printed on the printer. The printer is "listening."
PRINT #	PRINT #3 READY.		Use the PRINT # command to "unlisten" the printer.
CLOSE 3	CLOSE 3 READY.		You close the file so that <i>lfn</i> 3 can be used for something else.

## PRINTING UNDER PROGRAM CONTROL

As we have seen, you can control the printer directly from the keyboard. You can also control the printer from within a BASIC program. In the example below, this short BASIC program is placed in the computer's memory. (It could have been placed there from the keyboard, a cassette tape, or a floppy disk):

```
10 OPEN 3,4
20 PRINT #3, "THIS PROGRAM ILLUSTRATES PROGRAM CONTROL"
30 CLOSE 3
```

NO SPACE BETWEEN  
HERE

**REMEMBER THIS:** The LIST command within a program terminates program execution. Then, when you are finished running a program, you must type the PRINT # command to close the channel. Then type the CLOSE command to close the file.



The following example shows a BASIC program that takes advantage of some of your printer's special features. The program directs the printer to generate the following:

- An enhanced heading. You will find more about this feature on page 33.
- The entire printer character set.



Two special character functions are used in the program: the OFF/RVS key and the CRSR Down (Cursor Down) key. You may find out more about these starting on page 26.

A printout of a program you place in your computer's memory looks like this:


```
10 OPEN 4,4
20 FOR I = 32 TO 95                :A$=A$+CHR$(I):NEXT
30 FOR I = 160 TO 223            :B$=B$+CHR$(I):NEXT
40 C$=" CTRL RVS ON "+A$
50 D$=" CTRL RVS ON "+B$
60 E$=" CRSR DN "+A$
70 F$=" CRSR DN "+B$
80 G$=" CRSR DN "+C$
90 H$=" CRSR DN "+D$
100 PRINT #4,CHR$(14)"MODEL 1526 PRINTER CHARACTER SET"
110 PRINT #4:PRINT #4:PRINT #4
120 PRINT #4, A$
130 PRINT #4, B$
140 PRINT #4, C$
150 PRINT #4, D$
160 PRINT #4, E$
170 PRINT #4, F$
180 PRINT #4, G$
190 PRINT #4, H$
200 CLOSE 4
```

*No spaces  
between  
here*

Here is an explanation of each statement in the program.

Line No.	Explanation
10	Opens the printer as a file for printing.
20	Sets A\$ equal to a string containing all the unshifted characters from the keyboard.
30	Sets B\$ equal to a string containing all the shifted characters from the keyboard.
40	Sets C\$ equal to the set of all unshifted characters in reverse field. The character between the quotes is entered by pressing: "   ".
50	Sets D\$ equal to the set of all shifted characters in reverse field.
60-90	Each time a PRINT statement is encountered, as in lines 120-190, a Carriage Return is executed (unless a semicolon is placed at the end of the preceding PRINT statement). Therefore, each string, as in lines 160-190, must begin with a Cursor Down to reset the printer to lowercase.

NOTE: The use of a Cursor Down to set to lowercase and Cursor Up to reset to uppercase enables you to use upper- and lowercase letters on the same line.

60	Sets E\$ = to a Cursor Down plus A\$. The character between the quotes is entered by pressing: "  ".
70	Sets F\$ = to a Cursor Down plus B\$.
80	Sets G\$ = to a Cursor Down plus C\$.
90	Sets H\$ = to a Cursor Down plus D\$.
100-190	PRINT #4 tells the computer to print to device 4 (in this case, the printer).
100	CHR\$ (14) increases the width of the printed character to twice its original size. This line prints the title of the program in 'enhanced characters.'
110	Prints three Carriage Returns on the printer.
120-190	Prints each of the strings constructed in lines 20 through 90 on the printer.
200	Closes the printer channel.

NOTE: Printing a listing from within a program terminates the program. Therefore, LIST should be placed at the end of the program.



## FORMAT CONTROL

Through the printer's format control option, you can control the interpretation of data sent to the printer. The format control option allows you to print numbers in columns, set the number of lines per page, and perform other useful formatting tasks. To implement the format control option, you use the third parameter of the OPEN command.

### OPEN *lfn, dn, sa*

As before, *lfn* is the *logical file number*, *dn* is the *device number* (also known as the primary address), and *sa* is the *secondary address*. It is in the secondary address position that you enter the desired parameter.

You can enter any of the eleven secondary addresses in your OPEN command. These are:

- 0 Print data exactly as received in Upper/Graphics case.
- 1 Print data according to a previously-defined format
- 2 Store the formatting data
- 3 Set the number of lines per page to be printed
- 4 Enable the printer format diagnostic messages
- 5 Define a programmable character
- 6 Set spacing between lines
- 7 Print data exactly as received in Upper/Lower case.
- 9 Suppress diagnostic message printing
- 10 Reset printer

After the appropriate OPEN statement has been transmitted, a PRINT # statement is required to transmit the secondary address information to the specified device (in this case, your printer).

Remember that it is possible to have as many as ten files open simultaneously. This allows you to perform several formatting functions at one time on the data in the computer's memory. There are examples of the formatting capabilities described in this section in the Appendix.

## Printing Data Exactly As Received: *sa* = 0

This secondary address is the default value. Whether or not you include it in your OPEN statement, the printer prints data exactly as received. Up to 80 characters are printed on each line and if the next character is not a Carriage Return, a Carriage Return is performed automatically and the overflow characters are printed on the next line.

### Example:

```
10 OPEN 5,4
20 PRINT #5,"THIS IS A TEST!"
```

Results in:

```
THIS IS A TEST!
```

## Printing Data According to a Previously Defined Format: *sa* = 1

A secondary address of 1 invokes the formatting features of your printer. The data to be printed is arrayed according to a previously specified format using *sa* = 2. If you should transmit a string of data when *sa* = 1 is in effect and there is no formatting data in the printer's memory, then the data string is printed exactly as it is received.

When formatting string data from the computer, a skip, CHR\$(29), must be sent to delimit the end of a string being edited to a field. Leading blanks are stripped off a string; therefore, to print a blank alpha field you must transmit a shifted blank, CHR\$(160). The alpha field is then right padded with blanks as shown below.

### Example:

```
10 OPEN 2,4,2
20 OPEN 1,4,1
30 PRINT #2,"AAA  AAA  AAA"
40 PRINT #1,"ABC"CHR$(29)CHR$(160)CHR$(29)"DEF"
50 CLOSE 2:CLOSE 1
```

Results in:

```
ABC          DEF
```

Note: The data to be printed for any formatted line must be sent with only one "PRINT#" command.

## Storing the Formatting Data: sa = 2

Perhaps the most significant feature of your printer is its ability to format data. Picture formatting generates a simple one-to-one correspondence between the column position of the print line and the symbol that is to appear there. Formatting lets you left- or right-justify columns of data, or align numeric data on its decimal point position.

### Example:

```
10 OPEN 2,4,2
20 OPEN 1,4,1
30 PRINT #2,"$$$ .99"
40 PRINT #1,.05
50 CLOSE 2:CLOSE 1
```

Results in:

```
$.05
```

The commands in the example above transfer formatting data to the printer's internal memory for future print editing use. Once completed, READY and the blinking cursor appear on the screen.

## The Formatting Characters

Specification of a format is accomplished by a set of formatting characters which, when joined, define the rules of formatting data for the printer format interpreter. The formatting characters may be grouped into three classes:

Numeric	9,Z,\$,S,,-
Alpha	A
Skip	(Blank)

Fields are specified by combining these formatting characters. Up to 79 characters are accepted in a format string. Excess (overflow) characters are printed onto the next print line. Table 2 contains formatting examples.

## Numeric

9 - Specifies a digit position in a numeric field. If there is no digit to print in this position, then a blank is substituted.

Z - Also specifies a digit position in a numeric field. Unlike the 9, this character forces a 0 to be printed if there is no digit available for this position. This is desirable if leading zeroes are required in a numeric field.

\$ - if one \$ is specified, then the field is treated as a dollar amount with a fixed-sign position as follows:

\$123  
\$1234.00

If all digit positions to the left of the decimal point are \$, then the number is printed as a floating dollar sign right justified before the most significant digit.

S - When preceding a numeric field, the sign of the number (+ or -) is to be printed in this fixed column position.

. - Defines the position of the decimal point and is printed literally in this position.

— - Specifies a trailing sign. If the number is positive, a blank is printed. A numeric field cannot have both S and —. When this is the case, only the S will be honored.

Here are some examples of format fields:

- a) Field width and decimal position specified-  
99 Two-digit integer with no sign  
99.999 Five-digit real type with 3 positions to right of decimal
- b) Dollar sign insertion-  
\$99.99 Fixed dollar sign  
\$\$\$99 Floating dollar sign
- c) Leading zeros forced-  
ZZZZ No decimal  
ZZ.99 Decimal
- d) Trailing minus-  
999 - Blank is printed if number is positive
- e) Signed numbers-  
s99 + or - always printed

## Table 2. Formatting Examples

Table 2 contains more examples of sample data, format fields, and the resultant editing.

FORMAT FIELD	DATA	EDITED RESULT
AAAAA	ABC	ABC
AAAAA	ABCDEFG	ABCDE
\$\$\$\$	99	\$99
\$9999	99	\$ 99
\$99.99	77	\$77.00
\$99.99	- 77	\$77.00
\$99.99 -	- 77	\$77.00 -
\$99.99 -	77	\$77.00
S\$99.99	77	+ \$77.00
ZZZZ	77	0077
ZZ.999	77	77.000
ZZZ.99	77	077.00
999.99	77	77.00
.99	77	.**
.99	.001	.00
S.999	1.5E - 02	+ .015
Z.999 -	1.5E - 02	0.015
Z.999 -	- 1.5E - 02	0.015 -

The printer software handles up to ten significant figures, and an exponent range  $\pm 99$  on numbers passed as data to be formatted. Exponential numbers must be normalized such that  $0 \leq |X| < 10$ .

### Alpha (String Data):

The letter A represents one position of an alpha field. Within the field, leading blanks are truncated, the field is left-justified, and it is padded to the right with blanks. Shifted blanks CHR\$(160), are not deleted if in the leading positions.

#### Example:

```
10 OPEN 2,4,2
20 OPEN 1,4,1
30 PRINT #2, "A AA AAA"
40 PRINT #1, "CBM"CHR$(29) "CBM"CHR$(29) "CBM"
50 CLOSE 2:CLOSE 1
```

Results in:

```
C CB CBM
```



## Skip (Blank):

Simply use blanks where spaces are required:

### Example:

```
10 OPEN 2,4,2
20 OPEN 1,4,1
30 PRINT #2, "AAA AAA AAA"
40 PRINT #1, "PET"CHR$(29)"PET"CHR$(29)"PET"
50 CLOSE 2:CLOSE 1
```

Results in:

```
PET PET PET
```

To right justify integers, define an alpha string the length of the longest integer. Then blank fill the left of the integer data string and right justify the number within that string.

## Literals in Format Strings

Literals are characters that are to be printed exactly as they exit in the printer's memory rather than being used to format other data. Literals are flagged by preceding each literal in the format string with a reverse-field ON character.

One use of literal is to create a special form at the same time you are printing data. This is done by using the special graphics character to form vertical, horizontal, or other lines between the fields. You can create a form while printing data, either by overprinting graphic characters or by inserting literal characters in a format string as shown below.

The only limitation on literal characters in format strings is that they cannot be printed in the reverse field mode. They are also restricted to the same character set in which data is currently being printed.

### Example:

```
10 OPEN 2,4,2
20 OPEN 1,4,1
30 PRINT #2, " [CTRL RVS ON] [AAAAA [CTRL RVS ON] ] [CTRL RVS ON] (AAAAA [CTRL RVS ON] )
[CTRL RVS ON] ? AAAAA [CTRL RVS ON] ? "
32 A$=" "
35 FOR I=1 TO 10:A$=A$+CHR$(64+I)
40 PRINT #1, A$CHR$(29)A$CHR$(29)A$CHR$(29)
50 NEXT
60 CLOSE 2:CLOSE 1
```

Results in:

```
[ A ] ( A ) ?A ?
[AB ] (AB ) ?AB ?
[ABC ] (ABC ) ?ABC ?
[ABCD ] (ABCD ) ?ABCD ?
[ABCDE] (ABCDE) ?ABCDE?
[ABCDE] (ABCDE) ?ABCDE?
[ABCDE] (ABCDE) ?ABCDE?
[ABCDE] (ABCDE) ?ABCDE?
[ABCDE] (ABCDE) ?ABCDE?
[ABCDE] (ABCDE) ?ABCDE?
```

## Setting the Number of Lines Per Page: sa = 3

This special secondary address allows you to vary the number of printed lines per page. In order for this paging option to take effect, you must turn paging on with the special paging character, CHR\$(147), described on page 27. When paging is on and the paging secondary address is not implemented, the default number of lines per page is 66, including three blank lines at the top of the page and three blank lines at the bottom of the page.

### Example:

```
10 OPEN 4,4
20 OPEN 1,4,1
30 OPEN 2,4,2
40 OPEN 3,4,3
50 A$="999 9999 99.99999999 99.99999999"
55 PRINT #2,A$
60 PRINT #3, CHR$(60)
70 PRINT #4, CHR$(147)
80 FOR I = 1 TO 99
90 PRINT #1,I;I*I;SQR(I);I ↑ (1/3)
100 NEXT I
110 PRINT #4,CHR$(19)
120 CLOSE 4:CLOSE 3:CLOSE 2:CLOSE 1
```

## Enabling the Printer Diagnostic Messages: sa = 4

When a secondary address of 4 is transmitted, the printer's diagnostic messages appear whenever an error occurs. When a formatting error occurs, a message is printed, the format is dumped, and a pointer is located at the offending field. If a secondary address of 4 has NOT been transmitted, then once an error occurs, data characters are dumped directly to the print line even if the last command was to format data.

### Example:

Example:

```
*PE:C*
```

```
10 OPEN 4,4,4:PRINT #4:CLOSE 4
20 OPEN 25,4 25
30 PRINT #25
40 CLOSE 25
50 OPEN 1,4:CMD1:LIST
```

READY.

Whether or not diagnostic messages are enabled, overflow of numeric fields is always indicated by fields filled with asterisks (\*\*\*\*\*). This is a nonfatal error; as in other cases, the secondary address option is reset to zero and all data retrieved by the printer is printed exactly as received.

## Diagnostic Messages

- \*PE:L\* - Lines per page out of range. An attempt was made to set the lines per page to a value outside the range  $13 < \text{linesperpage} < 128$  via a print-to-SA 3. The command is ignored and the previous length remains in force.
- \*PE:C\* - Bad command. You gave the printer an invalid SA outside the range. The command is ignored.
- \*PE:M\* - Data-format mismatch. You gave the printer non-numeric data to print in a numeric field. The first character printed after the error message is the offending character.
- \*PE:E\* - Exponent error. The numeric data given to the printer to print in a numeric field had an invalid exponent. The expected form of an exponential number is:  $n.nnnn + ee$  or  $n.nnnn - ee$ . The expression must contain a plus or minus sign with a two digit exponent.
- \*PE:F\* - Bad format. The data that was sent to SA = 2 either contains illegal formatting characters, or unrecognizable syntax.
- \*PE:T\* - Terminator error. The SA changed before a terminator was detected. A terminator is defined as a carriage return ( $\text{chr}\$(13)$ ), a line feed ( $\text{chr}\$(10)$ ), or a carriage return line-feed sequence. Communication with the current SA must be terminated with a terminator before attempting to "talk" to another SA.

## Defining a Programmable Character: sa = 5

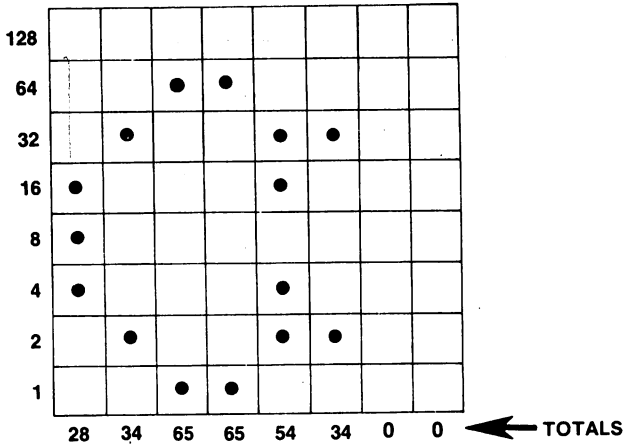
A secondary address of 5 allows you to create a custom character of your own. This programmable character is initialized with this secondary address.

Suppose you wanted to program the special symbol in Commodore Business Machines' logo, which is:



Lay out a 8x8 matrix (the same matrix as is on the print head). To the left of the matrix, write the binary bit value of each line. Use dots, one per square, to create your character. Then add up the binary bits indicated by your dots in each column. These totals are used in the DATA statement in your program.

Example:









The DATA statement in your program will read:

DATA 28,34,65,65,54,34,0,0

The program shown in the printout in the next example writes the Commodore logo ten times. It creates a string with the CHR\$ value of the column totals and passes the string to the printer with sa = 5. To achieve upper- and lowercase characters, use the CRSR Up (Cursor Up) for uppercase characters, and CRSR Down (Cursor Down) for lowercase characters.

Example:

```

10 DATA 28,34,65,65,54,34,0,0
20 OPEN 5,4,5
30 FOR I=1 TO 8:READ A:A$=A$+CHR$(A):NEXT
40 PRINT #5,A$
50 OPEN 4,4
60 FOR I=1 TO 10
70 PRINT #4,CHR$(14)CHR$(254)"  C  OMMODORE  B  BUSINESS
       M  ACHINES"
80 NEXT
90 CLOSE 5
100 CLOSE 4
    
```

The following is a description of the program shown above:

Line No.

20 Open 5,4,5

- Communicates to the printer what function you want executed. A secondary address of 5 indicates that you want something stored in the printer's custom character buffer.
- The device number (the printer's)
- The logical file number

30 The FOR NEXT loop contained in this line reads the data contained in line 10 and constructs a string A\$. A\$ contains the necessary information to print the Commodore logo.

40 Stores A\$ in printer memory.

50 Opens the print file for printing.

60-80 Prints the Commodore logo and "Commodore Business Machines" ten times.

- PRINT#4 Prints to the printer
- CHR\$(14) Prints 'enhanced characters'
- CHR\$(254) Prints the programmed character (the Commodore logo)
- Cursor Up Sets the printer to uppercase
- Cursor Down Sets the printer to lowercase

90-100 Closes the two files opened by the program.

After typing RUN, you get this result:

```

Q Commodore Business Machines
Q Commodore Business Machines
Q Commodore Business Machines
Q Commodore Business Machines
Q Commodore Business Machines
Q Commodore Business Machines
Q Commodore Business Machines
Q Commodore Business Machines
Q Commodore Business Machines
Q Commodore Business Machines

```

NOTE

Multiple programmable characters in the same line can only be made by overprinting. The programmable character cannot be changed when a line wraps around to the next line.

*Curly  
CMR 1.3.16*



## Suppress Diagnostic Message Printing: sa = 9

To turn off the diagnostic message printing, send a secondary address of 9.

### Example:

*PE:M*	COMMODORE
10 OPEN 4,4,4:PRINT #4:CLOSE 4	10 OPEN 4,4,4:PRINT #4:CLOSE 4
20 OPEN 2,4,2	15 OPEN 9,4,9:PRINT #9:CLOSE 9
30 OPEN 1,4,1	20 OPEN 2,4,2
40 PRINT #2,"\$\$\$\$.3"	30 OPEN 1,4,1
50 PRINT #1,"COMMODORE"	40 PRINT #2,"\$\$\$\$.9"
60 CLOSE 2:CLOSE 1	50 PRINT #1,"COMMODORE"
	60 CLOSE 2:CLOSE 1

## Resetting the Printer: sa = 10

To reset the printer, send a secondary address of 10.

### Example:

```
10 OPEN 10,4,10
20 PRINT #10
30 CLOSE 10
```

## SPECIAL CHARACTER FUNCTIONS

Special control characters can be used to change the mode of printing within a single line. Table 3 contains a summary of the special control characters. The table is followed by a brief description of each function.

### Enhanced Characters

You can print any character double its width. While characters are normally printed in a 8-row by 8-column matrix, enhanced characters appear in a 8-row by 16-column matrix. The example below shows how already enhanced characters can be further enhanced.

### Example:

```
10 OPEN 4,4
20 PRINT #4,"H"CHR$(14)"E"CHR$(14)"L"CHR$(14)"L"CHR$(14)"O"
30 CLOSE 4
```

Results in:

HELLO

## Paging

Paging must be turned on in your program with the code CHR\$(147) or the keyboard entry shown in Table 3. Otherwise, printing is continuous. The paging function provides for 66 lines per page including three blank lines at the top of the page and three blank lines at the bottom. The number of lines per page may be altered by using the secondary address 3 option. When paging is in effect, the paging off character performs a top-of-form function.

## Specifying Individual Upper- and Lowercase Characters

The printer default character set is ASCII/Graphics when power is first turned on to the printer. Though it is not possible to display both graphics and lowercase simultaneously on the video display screen, the printer does allow shifting between character sets on the same line. This function is analogous to the letters/figures shift on a Teletype™.

### Example:

```
10 OPEN 4,4
20 PRINT #4,"<CRSR UP> C <CRSR DOWN> OMMODORE"
```

Results in:

Commodore

## Reversing a Field

This function inverts the dot matrix to produce the effect of white on black (see Table 3). DO NOT use this mode of printing for more than five consecutive lines since extended printing in this mode will damage the print head.

### Example:

```
10 OPEN 5,4
20 PRINT #5,"   COMMODORE"
30 CLOSE 5
```

Results in:





**TABLE 3**  
**Special Control Character Summary**

Printer function	Code	ASCII	Keyboard
Enhance	CHR\$(14)	SO	NA
Unenhanced	CHR\$(129)		NA
Paging on	CHR\$(147)		SHIFT & CLR HOME
Paging off	CHR\$(19)	DC3	CLR HOME
RVS ON	CHR\$(18)	DC2	OFF RVS
RVS OFF	CHR\$(146)		SHIFT & OFF RVS
Carriage return	CHR\$(13)	CR	RETURN
Carriage return with no line feed	CHR\$(141)		NA
Line feed	CHR\$(10)	LF	NA
Uppercase	CHR\$(145)		↑ CRSR ↓    Cursor Up
Lowercase	CHR\$(17)		↑ CRSR ↓    Cursor Down
Skip space	CHR\$(29)		← CRSR →    Cursor Right
Quote	CHR\$(34)	"	"    Quote

### The Carriage Return

If you attempt to print more than 80 characters on a line, a Carriage Return with Line Feed will be forced and the overflow characters will be printed on the next line. If you transmit a Shift Carriage Return (\$8D), a Carriage Return without Line Feed is executed. This allows overprinting on a line. A Carriage Return turns off reverse field, character enhance and the quote mode.

### Quotation Marks

If an odd number of quotation marks have been transmitted, control characters are made visible. This can be particularly useful when you are making a listing of a BASIC program containing cursor control characters in quotation marks.

## APPENDIX

The programs in this appendix illustrate the many formatting capabilities of your printer. The first program is completely annotated. In the following programs, new concepts are commented upon.

### SIMPLE STRING FORMATTING

This program opens four files to perform four different tasks. It prints three lines: the format string, the unformatted data string, and the formatted data string.

Here is a printout from the program:

```
AA AA AA AA
ABC
AB AB AB AB
```

```
10 OPEN 1,4
20 OPEN 2,4,1
30 OPEN 3,4,2
40 OPEN 4,4,4
50 PRINT #4 :REM ENABLE ERROR DIAGNOSTICS
60 A$="AA AA AA AA" :REM FORMAT STRING
70 B$="ABC" :REM STRING TO BE FORMATTED
80 PRINT #1,A$
90 PRINT #1,B$
100 PRINT #3,A$
110 C$=CHR$(29)
120 PRINT #2,B$C$B$C$B$C$B$
130 CLOSE 1:CLOSE 2:CLOSE 3:CLOSE 4
```

An analysis of the program will help you understand how it works.

<b>Line No.</b>	<b>Explanation</b>		
10-40	Four logical files are opened to the printer so that the data in the computer's memory can be used in four different ways:		
	<b>Logical File No.</b>	<b>Secondary Address</b>	<b>Usage</b>
	1	0 (default)	Print data exactly as transmitted.
	2	1	Format data before it is printed.
	3	2	Transmit the format string.
	4	4	Enable format error diagnostic messages to be printed (if any).
50	Enables error messages (if any)		
60	Defines the format string		
70	Defines the data string		
80	Prints the format string		
90	Prints the unformatted data string		
100	Transmits the format string to the printer's memory		
110	Sets C\$ equal to the format character for "skip a space"		
120	Prints the formatted data string 4 times		
130	Closes the files		

# LEADING ZEROES FORCED

This program prints ten rows of numbers, each row beginning with a number one higher than the preceding row. In line 60, the formatting statement (which is prepared for in line 30, and transmitted in line 70) specifies leading zeroes in all fields and plus signs in the first two fields.

+0001.	+0002.	0003.	0004
+0002.	+0003.	0004.	0005
+0003.	+0004.	0005.	0006
+0004.	+0005.	0006.	0007
+0005.	+0006.	0007.	0008
+0006.	+0007.	0008.	0009
+0007.	+0008.	0009.	0010
+0008.	+0009.	0010.	0011
+0009.	+0010.	0011.	0012
+0010.	+0011.	0012.	0013

```
10 OPEN 1,4
20 OPEN 2,4,1
30 OPEN 3,4,2
40 OPEN 4,4,4
50 PRINT #4 :REM ENABLE ERROR DIAGNOSTICS
60 F$="SZZZZ. SZZZZ. ZZZZ. ZZZZ"
70 PRINT #3,F$
80 FOR I = 1 TO 10
90 PRINT #2,I,I+1,I+2,I+3
100 NEXT
110 CLOSE 1:CLOSE 2:CLOSE 3:CLOSE 4
```

## SIGNED NUMBERS

As shown below, a trailing sign, when specified in a format statement, either prints a blank for a positive number, or a minus (-) for a negative number. This form is sometimes desired by accountants.

A leading sign always prints a plus (+) or a minus (-) for the sign of the number.

If no sign is specified in the format, then no sign is printed. This works essentially like an absolute value on the number.

1   -   1   1-

```
10 OPEN 1,4
20 OPEN 2,4,1
30 OPEN 3,4,2
40 OPEN 4,4,4
50 PRINT #4          :REM ENABLE ERROR DIAGNOSTICS
60 F$="999  S999  999-"
70 PRINT #3,F$
80 PRINT #2,-1,-1,-1
90 CLOSE 1:CLOSE 2:CLOSE 3:CLOSE 4
```

## DECIMAL FRACTION SPECIFICATION

When the right hand side of the decimal point in a format statement has more digits of precision specified than contained in the formatted data, the remaining digit positions are filled with zeroes. If the reverse is true, then the formatted number is truncated.

5.00	-	1.25000	3
10.00	-	2.50000	6
15.00	-	3.75000	9
20.00	-	5.00000	12
25.00	-	6.25000	15
30.00	-	7.50000	18
35.00	-	8.75000	21
40.00	-	10.00000	24
45.00	-	11.25000	27
50.00	-	12.50000	30

```
10 OPEN 1,4
20 OPEN 2,4,1
30 OPEN 3,4,2
40 OPEN 4,4,4
50 PRINT #4      :REM ENABLE ERROR DIAGNOSTICS
60 F$="999.99  S999.99999  999 "
70 PRINT PRINT #3,F$
80 FOR I=1 TO 10
90 PRINT #2,-10*I/2,-5*I/4,3*I
100 NEXT
110 CLOSE 1: CLOSE 2: CLOSE 3: CLOSE 4
```

## FIXED AND FLOATING \$ FIELDS

In fixed dollar fields the dollar sign (\$) is printed in the leftmost position and leading unused digit positions are printed as blanks.

In floating dollar fields the \$ is printed immediately to the left of the most significant figure or decimal point. Unused digit fields to the left of the \$ are printed blank.

\$66.29	\$6.62	\$ 6629	\$.66
\$45.83	\$4.58	\$ 4583	\$.45
\$32.86	\$3.28	\$ 3286	\$.32
\$53.33	\$5.33	\$ 5333	\$.53
\$74.91	\$7.49	\$ 7491	\$.74
\$ 7.79	\$.77	\$ 779	\$.07
\$97.51	\$9.75	\$ 9751	\$.97
\$27.03	\$2.70	\$ 2703	\$.27
\$15.28	\$1.52	\$ 1528	\$.15
\$49.95	\$4.99	\$ 4995	\$.49

```
10 OPEN 1,4
20 OPEN 2,4,1
30 OPEN 3,4,2
40 OPEN 4,4,4
50 PRINT #4
60 F$="99.99    $$$ .99    $99999    $.99"
70 PRINT#3,F$
80 FORI=1 TO 10
85 X=RND(1)
90 PRINT#2,X*100,X*10,X*10000,X
100 NEXT
110 CLOSE1:CLOSE2:CLOSE3:CLOSE4
```

READY.

## MIXED FORMAT FIELD TYPES

Shown below are an integer with leading zeros, alpha field, floating dollar, and an integer with zero suppressed.

In line 120, CHR\$(29), (skip) is used as a terminator for the alpha field so that it is formatted correctly.

```
00002    PET COMPUTER                $795.00    50

10 OPEN 1,4
20 OPEN 2,4,1
30 OPEN 3,4,2
40 OPEN 4,4,4
50 PRINT #4           :REM ENABLE ERROR DIAGNOSTICS
60 F$="ZZZZZ  AAAAAAAAAAAAAAAAAAAAAA  $$$$$.99  999"
70 PRINT #3,F$
80 A=2
90 B$="PET COMPUTER"
100 C=795
110 D=50
120 PRINT #2,A,B$,CHR$(29),C,D
130 CLOSE 1: CLOSE 2: CLOSE 3: CLOSE 4
```



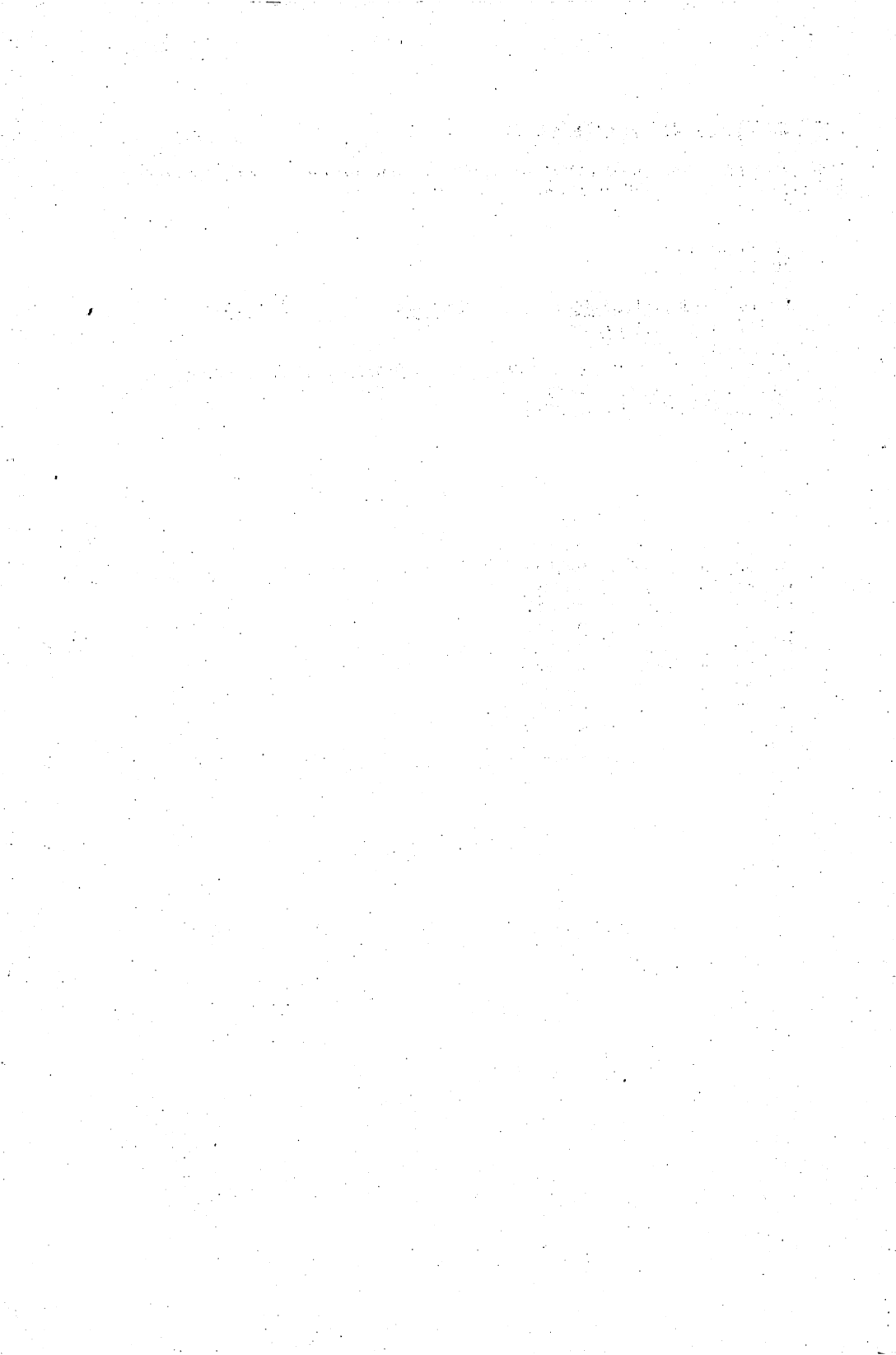
## FORMATTING WITH LITERALS

In the example below, literal characters within a format line are created by preceding each literal character with the reverse on symbol.

```
10 OPEN 1,4
20 OPEN 2,4,2
30 OPEN 3,4,1
40 F$="CTRL RVS | CTRL RVS * ZZ CTRL RVS ◆ $$$$ CTRL RVS |
   ZZ.999 CTRL RVS +"
50 PRINT#2,F$
60 FOR I=1 TO 10:X=10*RND(1):Y=1000*RND(1):Z=8*RND(1)
70 PRINT#3,X;Y;Z:NEXT
80 CLOSE1:CLOSE2:CLOSE3

READY.
```

```
|* 04 ◆ $502 | 03.756 +
|* 08 ◆ $582 | 02.813 +
|* 04 ◆ $7 | 00.268 +
|* 04 ◆ $670 | 00.969 +
|* 03 ◆ $272 | 07.298 +
|* 03 ◆ $292 | 00.234 +
|* 04 ◆ $382 | 03.260 +
|* 06 ◆ $76 | 03.901 +
|* 07 ◆ $73 | 01.361 +
|* 07 ◆ $645 | 03.354 +
```



## **COMMODORE SALES CENTERS**

### **Commodore Business Machines, Inc.**

1200 Wilson Drive  
Westchester, PA 19380, U.S.A.

### **Commodore Business Machines Ltd.**

3370 Pharmacy Avenue, Agincourt,  
Ontario, M1W 2K4, Canada

### **Commodore Business Machines (UK) Ltd.**

675 Ajax Avenue, Trading Estate,  
Slough Berks, SL1 4BG, England

### **Commodore Bueromashinen GmbH**

Lyonerstrasse 38, PO BOX 710126  
Frankfurt 6000, West Germany

### **Commodore Italiana S.R.L.**

Via Conservatoria 22  
Milano 20122, Italy

### **Commodore Information Center**

5 Onion Road  
Lane Cove 2066 NSW, Australia

### **Commodore Computer B.V.**

Marksingel 2e4811 N.V. Breda  
Postlrus 720, 480 3aS Breda, Netherlands

### **Commodore AG (Schweiz)**

Aeschenvorstadt 57  
4010 Basel, Switzerland

## DISK DRIVES

### MODEL 1541

Media compatible with 4040 and 2031LP diskettes.  
170 K of memory per single side disk.  
Serial port interface.

### MODEL SFD 1001

1 Megabyte double sided floppy disk drive.  
Uses double density, double sided diskettes.  
IEEE interface.

### MODEL 2031LP

Media compatible with 1541 and 4040 diskettes.  
170 K of memory per single side disk.  
Rugged and stylish housing.  
IEEE interface.

## PRINTERS

### MODEL MPS-801

High quality dot matrix printing.  
80 column printing.  
Prints 50 characters per second.  
Serial port interface.

### MODEL MCS-801

Seven color dot matrix printing.  
Up to 80 columns printed per page.  
Prints 38 characters per second.  
Perfect for everything from personal letters to important business reports.  
Serial port interface.

## PRINTER/PLOTTER

### MODEL 1520

Four color printer/plotter for all graphic needs.  
Print bar graphs... create pie charts... plot time graphs.  
High quality printing in a variety of styles and sizes.  
Great for graphic artists.

## MODEMS

### VICMODEM

Timely financial information, news and reference libraries are as near as your phone.  
An affordable introduction to the growing world of phone/computer services.  
Communicate with a wide variety of computer users who already own a modem.

### AUTOMODEM

The convenience of automatic answering and dialing.  
The resource power of telecomputing services.  
Turn your phone into a telecomputing information network.

 **commodore**  
COMPUTER